Statements DISPLAY and WRITE

This document describes how to use the statements DISPLAY and WRITE to output data and control the format in which information is output.

The following topics are covered:

DISPLAY Statement

The DISPLAY statement produces output in column format; that is, the values for one field are output in a column underneath one another. If multiple fields are output, that is, if multiple columns are produced, these columns are output next to one another horizontally.

The order in which fields are displayed is determined by the sequence in which you specify the field names in the DISPLAY statement.

The DISPLAY statement in the following program displays for each person first the personnel number, then the name and then the job title:

```
** Example Program 'DISPLX01'
DEFINE DATA LOCAL
1 VIEWEMP VIEW OF EMPLOYEES
  2 PERSONNEL-ID
  2 NAME
  2 BIRTH
  2 JOB-TITLE
END-DEFINE
READ (3) VIEWEMP BY BIRTH
  DISPLAY PERSONNEL-ID NAME JOB-TITLE
END-READ
END
```

```
Page      1                                                  99-01-22  11:31:01

PERSONNEL         NAME                 CURRENT
   ID                                  POSITION
--------- -------------------- -------------------------

30020013  GARRET               TYPIST
30016112  TAILOR               WAREHOUSEMAN
20017600  PIETSCH              SECRETARY
```


To change the order of the columns that appear in the output report, simply reorder the field names in the DISPLAY statement. For example, if you prefer to list employee names first, then job titles followed by personnel numbers, the appropriate DISPLAY statement would be:

```
** Example Program 'DISPLX02'
DEFINE DATA LOCAL
1 VIEWEMP VIEW OF EMPLOYEES
  2 PERSONNEL-ID
  2 NAME
  2 BIRTH
  2 JOB-TITLE
END-DEFINE
READ (3) VIEWEMP BY BIRTH
  DISPLAY NAME JOB-TITLE PERSONNEL-ID
END-READ
END
```

```
Page      1                                                  99-01-22  11:32:06

        NAME                  CURRENT           PERSONNEL
                             POSITION             ID
-------------------- ------------------------- ---------

GARRET               TYPIST                    30020013
TAILOR               WAREHOUSEMAN              30016112
PIETSCH              SECRETARY                 20017600
```

A header is output above each column. Various ways to influence this header are described in the document Column Headers.

WRITE Statement

The WRITE statement is used to produce output in free format (that is, not in columns). In contrast to the DISPLAY statement, the following applies to the WRITE statement:

- If necessary, it automatically creates a line advance; that is, a field or text element that does not fit onto the current output line, is automatically output in the next line.
- It does not produce any headers.
- The values of a multiple-value field are output next to one another horizontally, and not underneath one another.

The two example programs shown below illustrate the basic differences between the DISPLAY statement and the WRITE statement.

You can also use the two statements in combination with one another, as described later in the document Vertical Displays, Combining DISPLAY and WRITE.

Example of DISPLAY Statement

```
** Example Program 'DISPLX03'
DEFINE DATA LOCAL
1 VIEWEMP VIEW OF EMPLOYEES
  2 NAME
  2 FIRST-NAME
  2 SALARY (1:3)
END-DEFINE
```



```
READ (2) VIEWEMP BY NAME STARTING FROM 'JONES'
  DISPLAY NAME FIRST-NAME SALARY (1:3)
END-READ
END
```

```
Page      1                                                  97-08-14  11:44:00

        NAME              FIRST-NAME        ANNUAL
                                            SALARY
-------------------- -------------------- ----------

JONES                VIRGINIA                  46000
                                               42300
                                               39300
JONES                MARSHA                    50000
                                               46000
                                               42700
```

Example of WRITE Statement

```
** Example Program 'WRITEX01'
DEFINE DATA LOCAL
1 VIEWEMP VIEW OF EMPLOYEES
  2 NAME
  2 FIRST-NAME
  2 SALARY (1:3)
END-DEFINE
READ (2) VIEWEMP BY NAME STARTING FROM 'JONES'
  WRITE NAME FIRST-NAME SALARY (1:3)
END-READ
END
```

```
Page      1                                                  97-08-14  11:45:00

JONES                VIRGINIA                46000      42300      39300
JONES                MARSHA                  50000      46000      42700
```

Column Spacing - SF Parameter and *n*X Notation

By default, the columns output with a DISPLAY statement are separated from one another by **one** space.

With the session parameter SF, you can specify the default number of spaces to be inserted between columns output with a DISPLAY statement. You can set the number of spaces to any value from 1 to 30.

The parameter can be specified with a FORMAT statement to apply to the whole report, or with a DISPLAY statement at statement level, but not at field level.

With the *n*X notation in the DISPLAY statement, you can specify the number of spaces (*n*) to be inserted between two columns. An *n*X notation overrides the specification made with the SF parameter.


```
** Example Program 'DISPLX04'
DEFINE DATA LOCAL
1 VIEWEMP VIEW OF EMPLOYEES
  2 PERSONNEL-ID
  2 NAME
  2 BIRTH
  2 JOB-TITLE
END-DEFINE
FORMAT SF=3
READ (3) VIEWEMP BY BIRTH
  DISPLAY PERSONNEL-ID NAME 5X JOB-TITLE
END-READ
END
```

The above example program produces the following output, where the first two columns are separated by 3 spaces due to the SF parameter in the FORMAT statement, while the second and third columns are separated by 5 spaces due to the notation "5X" in the DISPLAY statement:

```
Page      1                                                  99-01-22  11:33:40

PERSONNEL           NAME                     CURRENT
   ID                                        POSITION
---------   --------------------     -------------------------

30020013    GARRET                   TYPIST
30016112    TAILOR                   WAREHOUSEMAN
20017600    PIETSCH                  SECRETARY
```

The *n***X** notation is also available with the WRITE statement to insert spaces between individual output elements:

```
WRITE PERSONNEL-ID 5X NAME 3X JOB-TITLE
```

With the above statement, 5 spaces will be inserted between the fields PERSONNEL-ID and NAME, and 3 spaces between NAME and JOB-TITLE.

Tab Setting - *n*T Notation

With the *n*T notation, which is available with the DISPLAY and the WRITE statement, you can specify the position where an output element is to be output.

```
** Example Program 'DISPLX05'
DEFINE DATA LOCAL
1 VIEWEMP VIEW OF EMPLOYEES
  2 NAME
  2 FIRST-NAME
END-DEFINE
READ (3) VIEWEMP BY NAME STARTING FROM 'JONES'
  DISPLAY 5T NAME 30T FIRST-NAME
END-READ
END
```

The above program produces the following output, where the field NAME is output starting in the 5th position (counted from the left margin of the page), and the field FIRST-NAME starting in the 30th position:


```
Page      1                                                  97-08-21  11:46:01

             NAME                  FIRST-NAME
    --------------------     --------------------
             JONES                 VIRGINIA
             JONES                 MARSHA
             JONES                 ROBERT
```

Line Advance - Slash Notation

With a slash "/" in a DISPLAY or WRITE statement, you cause a line advance.

- In a DISPLAY statement, a slash causes a line advance *between fields* and *within text*.
- In a WRITE statement, a slash causes a line advance only when placed *between fields*; within text, it is treated like an ordinary text character.

When placed between fields, the slash must have a blank on either side.

For multiple line advances, you specify multiple slashes.

Example of Line Advance in DISPLAY Statement

```
** Example Program 'DISPLX06'
DEFINE DATA LOCAL
1 VIEWEMP VIEW OF EMPLOYEES
  2 NAME
  2 FIRST-NAME
  2 DEPARTMENT
END-DEFINE
READ (3) VIEWEMP BY NAME STARTING FROM 'JONES'
  DISPLAY NAME / FIRST-NAME 'DEPART-/MENT' DEPARTMENT
END-READ
END
```

The above DISPLAY statement produces a line advance after each value of the field NAME and within the text "DEPART-MENT":

```
Page      1                                                  97-08-14  11:45:12

          NAME         DEPART-
       FIRST-NAME       MENT
  -------------------- -------

  JONES                SALE
  VIRGINIA
  JONES                MGMT
  MARSHA
  JONES                TECH
  ROBERT
```

Example of Line Advance in WRITE Statement

```
** Example Program 'WRITEX02'
DEFINE DATA LOCAL
1 VIEWEMP VIEW OF EMPLOYEES
  2 NAME
  2 FIRST-NAME
  2 DEPARTMENT
END-DEFINE
READ (3) VIEWEMP BY NAME STARTING FROM 'JONES'
  WRITE NAME / FIRST-NAME 'DEPART-/MENT' DEPARTMENT //
END-READ
END
```

The above WRITE statement produces a line advance after each value of the field NAME, and a double line advance after each value of the field DEPARTMENT, but none within the text "DEPART-/MENT":

```
Page      1                                                  97-08-14  11:45:12

JONES
VIRGINIA             DEPART-/MENT SALE


JONES
MARSHA               DEPART-/MENT MGMT


JONES
ROBERT               DEPART-/MENT TECH
```

Further Examples of DISPLAY and WRITE Statements

See the following example programs in library SYSEXPG:

- DISPLX13
- WRITEX08
- DISPLX14
- WRITEX09
- DISPLX21

